International Journal for Housing Science and Its Applications
Publish August 5, 2025. Volume 46, Issue 3 Pages 4035-4043

https://doi.org/10.70517/ijhsa463343

Research on Cloud Data Integrity Verification Technology Based on Big Data Analysis in Intelligent Accounting Cloud Platform

Yuhua Chen[1,2,*], Hasri Mustafa[2], Asna Atqa Abdullah[2] and Ziqin Feng[3]
[1] School of Finance and Taxation, Zhengzhou Technology and Business University, Zhengzhou, Henan, 450000, China
[2] School of Business and Economics, University Putra Malaysia, Serdang, 43400, Malaysia
[3] School of Management and Economics, North China University of Water Resources and Electric Power, Zhengzhou, Henan, 450000, China
Corresponding authors: (e-mail: chenyuhua25218@163.com).

Abstract Aiming at the problem of accounting information security under the financial co-working mode, this paper proposes a data integrity verification algorithm based on big data analysis. The intelligent accounting cloud platform based on cloud computing is analyzed to build the overall system architecture and the network topology of cloud storage data. A single-audit scheme based on the encryption-evidence chain algorithm is proposed, which uses the ZSS short signature algorithm to calculate the labels of data chunks, and improves the verification efficiency through bilinear pairs and generalized cryptographic hash functions. Simulation tests and experimental results based on the AliCloud platform show that all five algorithms can complete the corresponding data processing work in a short time. When verifying 500 pieces of data at the same time, the total elapsed time of the ZSS short signature scheme after the introduction of bilinear pairs and generalized cryptographic hash function is 14753.96ms, which reduces the elapsed time by 50.66% compared with the original scheme. The average verification time consumed in ten experiments with data sizes of 1MB, 50MB, 500MB, and 1000MB is 601.12s, 644.73s, 987.17s, and 1267.32s, respectively, and the time trend consumed by the TPSP to generate the hash value is basically the same as that of the verification time, which proves that the scheme in this paper is safe and efficient.

Index Terms accounting cloud platform, data integrity verification, single audit scheme, bilinear mapping, ZSS short signature technique

I. Introduction

Driven by the computerization of accounting and the sharing economy, the financial sharing model has gradually developed into an effective way for many large and medium-sized enterprises to improve their financial management methods, innovate their enterprise management models, improve their management level, and reduce their operating costs [1]-[3]. In this context, more and more enterprises and individuals choose to access cloud computing services to save their data and files on cloud servers [4]. Unlike local storage, users can greatly improve work efficiency and reduce hardware investment costs by using cloud data [5]. Based on the intelligent accounting cloud platform, enterprises will import their own financial control information data into the network and realize resource sharing in the network cloud [6], [7]. Enterprise accounting staff can use personal computers, smart phones and other network terminals, without time and space constraints, to deal with relevant information and business content at any time and anywhere, thus greatly improving the flexibility and effectiveness of the work [8]-[11].

At the same time, the enterprise management staff can through the cloud accounting in a timely and rapid collection of financial data information in the enterprise, will be able to effectively integrate a variety of data, and on this basis for in-depth analysis of data information and mining, so as to carry out a full range of enterprise financial management work [12]-[14]. But the combination of accounting informationization and cloud computing will also face some new problems. Hackers may try to tamper with the user data on the cloud storage, in addition, data loss is inevitable in the process of system operation [15], [16]. When something similar to the above occurs, the cloud platform provider needs to detect the anomaly in time in order to deal with it as soon as possible. Therefore, the data on the cloud server must be constantly verified to ensure the consistency and integrity of the data storage files [17].

This paper firstly describes the theory of cloud computing and related technologies, and utilizes cloud computing technology and SOA mode to construct an intelligent accounting informatization system. A single-audit scheme for data integrity verification is designed to solve the problem of information leakage by using ZSS short signature

technology based on bilinear mapping. The security analysis is developed from three dimensions to verify the reliability of the scheme. Simulation tests are conducted based on the AliCloud platform to analyze the operation efficiency of the proposed algorithm. Relying on the total elapsed time comparison, test the effect of the improved scheme in this paper. Demonstrate the performance level of cloud data integrity verification of this paper's scheme through four different sizes of data test experiments.

II. Design of Cloud Data Integrity Verification Scheme in Smart Accounting Cloud Platform

Under the impetus of accounting computerization and sharing economy, the financial sharing mode has gradually developed into an effective way for many large and medium-sized enterprises to improve financial management, innovate enterprise management mode, improve management level and reduce enterprise operation cost, and the combination of accounting informatization and cloud computing has become a hot direction for research. However, the combination of accounting informatization and cloud computing will also face some new problems. Hackers may try to tamper with the user data on the cloud storage, in addition, data loss is inevitable during the system operation, if the above similar situation occurs, the cloud platform provider needs to find out the abnormality in time in order to deal with it as soon as possible. Therefore, the data on the cloud server must be constantly verified to ensure the consistency and integrity of the data storage files. In order to effectively improve the efficiency of integrity verification, this paper proposes a cloud data integrity verification algorithm based on big data analysis.

II. A.Intelligent accounting informatization system based on cloud computing

II. A. 1) Cloud Computing Theory and Related Technologies

Cloud computing is currently a research hotspot in computer science and technology, has been the attention of many enterprises and institutions and related Internet experts, is an important trend in the future development of computer network technology.

A typical cloud computing platform needs to have: gridded data storage matrix network, firewall equipment, computing resources equipment, and allows users to use a scalable cloud storage space remotely by leasing, to achieve cloud application services, cloud computing service principle shown in Figure 1.

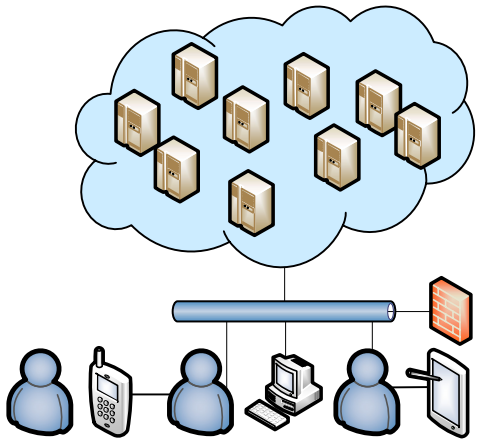

Figure 1: Principle of cloud computing service

A complete cloud computing architecture should include: access layer, core layer, resource aggregation layer, API interface layer and application layer.

II. A. 2) Overall architecture of the information technology platform

The combination of accounting informatization and cloud computing can effectively enhance financial shared management, which can greatly improve work efficiency and reduce hardware investment costs. The financial management (accounting, finance) service system constructed by using cloud computing technology and SOA mode can realize the enterprise to obtain shared resources and reduce the cost of enterprise informatization. Based on the research content of the above literature, this paper provides a specific design for the overall architecture of the cloud computing-based accounting informatization platform, including five modules: process management module, SAP module, file management module, procurement management module, contract management module, and the overall architecture is shown in Figure 2.

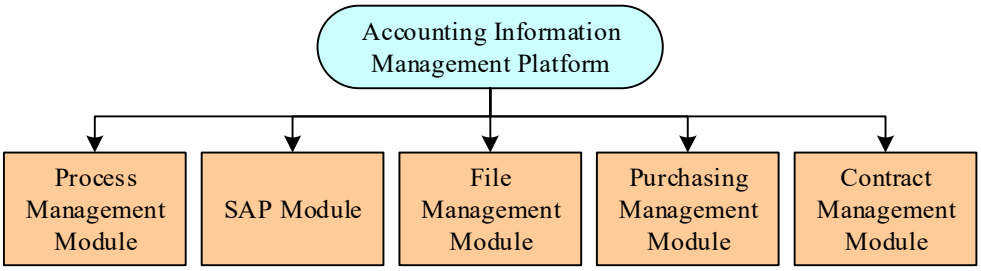

Figure 2: Overall functional architecture of informationization platform

II. B.Single audit program design

II. B. 1) Main ideas

(1) Separate storage of labels and file chunking data: data files are chunked and corresponding data labels are generated, the chunks are stored to the cloud server, the T-Merkle hash tree is generated using the labels and stored to the blockchain.

(2) Verification of integrity using ZSS short signature scheme: when verifying the integrity, the auditor takes the evidence calculated based on the data entities from the server, takes the data labels calculated during initialization from the blockchain, and finally calculates the correctness of the verification evidence to draw a conclusion.

(3) Implementing data update operation by appending: tagging the type of dynamic operation, CS and blockchain determine the corresponding operation according to the type.

(4) Implementing batch verification tasks by aggregating proofs: we utilize the stronger computing power of cloud servers and blockchain to aggregate labeled proofs and data proofs separately, instead of the auditor performing the aggregation operation. The auditor only verifies the final proof results.

II. B. 2) Definitions and frameworks

The scheme proposed in this paper consists of five main algorithms:

(1) $KeyGen(\lambda) \to (sk, pk)$. Key generation algorithm, the input is the system parameter λ and the output is the public key pk and the private key sk.

(2) $TagGen(F, sk) \to (F_{\inf o}, T)$. Tag generation algorithm with inputs of the original file and the private key sk, and outputs of other information such as the file chunking and index collection, and the tag collection T.

(3) $ChallGen(F_{\inf o}) \to Q$. Questioning information generation algorithm, the input is the file data block index and the output is the questioning set Q.

(4) ProofGen algorithm consists of TPProofGen and DPProofGen:

$TPProofGen(tag, Q) \to TP$. Tag proof generation algorithm with input the set of tags tag and the set of queries Q, and output the tag proof TP,

$DP\,\mathrm{Pr}\,oofGen(f, Q) \to DP$. Data Proof Generation Algorithm, where the input is the data chunking and questioning set Q and the output is the data proof DP.

(5) $Verify(pk, Q, TP, DP) \to (true / false)$. The inputs to the verification algorithm are the public key pk, the question set, the labeling evidence and the data evidence, and the outputs are $true / false$.

The audit framework of the program proposed in this paper is shown in Fig. 3:

System initialization phase: (1) Users locally execute the $KeyGen$ algorithm to generate public-private keys. (2) The user executes $TagGen$ locally to chunk the file and generate the tags for the chunks. (3) The user uploads the data chunks and tag set to the cloud server and blockchain respectively.

Integrity verification phase: (1) The auditor executes the algorithm $ChallGen$ to generate the query Q, and the query chunks are the result of random sampling. Send Q to the cloud server and blockchain respectively. (2) Upon receipt of Q, the cloud server exec $DPProofGen$ generates the data proof DP, and the blockchain exec $TPProofGen$ generates the tag proof TP and sends it to the auditor. (3) After receiving DP and TP, the auditor executes the $Verify$ algorithm and the result is $true / false$. $true$ proves that the data is complete, otherwise the data is corrupted or lost.

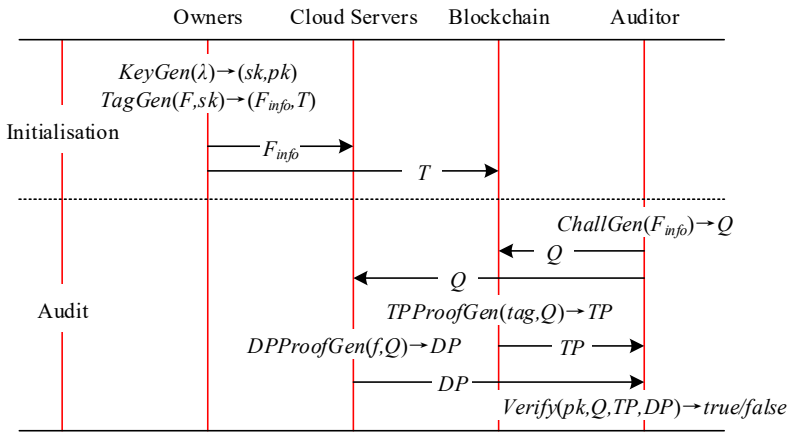

Figure 3: Audit framework

II. B. 3) Program details

In the scheme of this paper, data labels are computed by ZSS short signatures and stored on the blockchain.The ZSS scheme uses bilinear pairs and generic cryptographic hash functions (e.g., SHA-1 or MD5) without the need for special hash functions (e.g., BLS). In addition, it is faster than BLS signatures because the verification process requires fewer pairing operations. The scheme consists of two phases: an initialization phase and a verification phase.

(1) System initialization phase

First, the client (Client) generates a random value $sk \in Z_p$ as the labeled private key and computes the public key $pk = skP$ from the private key. Under the Inv-CDHP (Inverse Computational Diffie-Hellman Problem) assumption, it is impossible to derive the private key from the public key.

Second, the client splits the encrypted data file F into n blocks, denoted as $F = \{m_1, m_2, \ldots m_n\}$, and then computes a data label Tag_i for each block m_i as:

$$Tag_i = \frac{1}{H(m_i) + sk} P \tag{1}$$

where H is a generic hash function, such as MD-5 or SHA-1.The set of labels for the data file F is denoted as $T = \{Tag_1, Tag_2, \cdots Tag_n\}$.

Finally the client will outsource the data file F to a cloud storage service provider and upload the collection of data tags T to the blockchain. After the upload, the client will be able to delete or retain the original data with the data tags. The data tags are packaged into blockchain nodes on the blockchain through the T-Merkle hash tree. In addition, the cloud storage service provider will do a block integrity verification before receiving the data to prevent malicious clients. The verification process is similar to the validation phase described below.

(2) Integrity verification phase

The client (as the verifier) selects a set of random elements $I = \{s_1, s_2, \ldots s_c\}$, where $c \in [1, n]$. A pseudo-random value u_i is then generated for each s_i in Z_p. The client sends queries $Chall = \{i, u_i\}_{i \in I}$ to the blockchain and the cloud server, respectively, and the cloud server receives the query message, Chall, and computes the data proof DPs by bilinear mapping encryption:

$$DP = e\left(\sum_{i \in I} u_i H(m_i) P, P\right) \tag{2}$$

Meanwhile, the blockchain finds the questioned tag and then calculates the cryptographic tag proof according to the following formula:

$$TP = e\left(\sum_{i \in I} \frac{u_i}{Tag_i} P^2, P\right) \tag{3}$$

The cloud server and the blockchain return $\{DP, TP\}$ to the client as a proof of verification information, respectively.

After receiving the proof of data DP and the proof of label TP, the client first uses the generated $Chall = \{i, u_i\}_{i \in I}$ and the public key pk to compute $R = \sum_{i \in I} u_i pk$, and then verifies the following equation by verifying the following to verify the plain:

$$TP = DP \cdot e(R, P) \tag{4}$$

If Equation (4) holds, the data file on the cloud storage server is intact. Otherwise, it means that the file is corrupted.

II. C. Security analysis

II. C. 1) Audit correctness

If the CSP stores the DO's data correctly, the verifier can validate the proof information it generates by using the validation algorithm. Here the validation is carried out using the left equation pushed to the right, and the validation equation is as follows:

$$\begin{aligned} e(\sigma, g) &= e\left(\prod_{i=s_1}^{s_c} \left(H(w_i) \cdot m_i\right)^{xv_i}, g\right) \\ &= e\left(\prod_{i=s_1}^{s_c} \left(H(w_i) \cdot m_i\right)^{v_i}, g^x\right) \\ &= e\left(\prod_{i=s_1}^{s_c} \left(H(w_i)^{v_i} \cdot m_i^{v_i}\right), v\right) \\ &= e\left(\mu \cdot \prod_{i=s_1}^{s_c} \left(H(w_i)\right)^{v_i}, v\right) \end{aligned} \tag{5}$$

II. C. 2) Privacy

Since the CSP generates the proof information $P = \{\mu, \sigma, \tau\}$, where τ refers to the common parameters of the system, independent of the data m_i. $\mu = \Pi_{j \in I} m_j^{v_j}$ and $\sigma = \Pi_{j \in I} \sigma_j^{v_j}$, where μ and σ refer to the aggregation of the data blocks by the corresponding random values, not the linear combination of data, the validator cannot obtain valid information about DO from μ and σ in the audit message. The verifier also cannot recover m_i from the proof information P of the CSP response, thus realizing data privacy protection.

II. C. 3) Non-interactivity

In traditional auditing programs, validation is performed by using a challenge-response approach, where the TPA or DO generates a challenge message and sends it to the CSP to interact with the validator. The CSP then generates the appropriate proof information based on the challenge message and returns it to the verifier for verification. This scheme does not adopt the challenge-one-response approach, the verifier does not need to generate the challenge message, and the CSP does not need to interact with the verifier.The CSP generates the challenge message with the challenge message by obtaining the current state information from the public information in the blockchain, which is out of anyone's control and unpredictable. Therefore, the generated challenge messages cannot be controlled by anyone and are randomized, which ensures the randomness of challenge message generation. In the audit phase, TPA first verifies the accuracy of the public information and generates challenge messages based on the public information for verification.

III. Simulation test for integrity verification of intelligent accounting cloud platform based on big data analysis

In order to verify the effectiveness of the proposed algorithm in this paper, the paper is based on the AliCloud platform for simulation testing. In this paper, personal medical record data is used as the data to be encrypted and stored, which are unstructured data containing text and images, and the size of each data sample is 1MB.

III. A. Algorithm validity test

In this paper, different amounts of data are used to test the time required by the algorithms of this paper for data coding and decoding, and the results are shown in Fig. 4. From Fig. 4, it can be seen that all five algorithms are able to complete the corresponding data processing work in a shorter time, and the algorithm processing time is kept below 30s when the data sample length is 50.

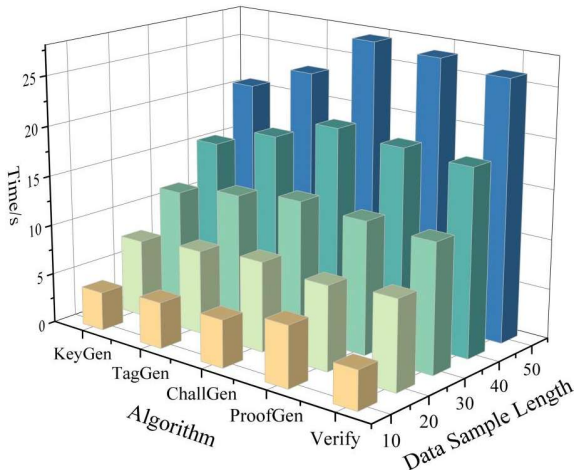

Figure 4: Processing time for data of different lengths

In this paper, the data recovery ability of this paper's algorithm is tested by adding different lengths of redundant data, and the coding results of different lengths of data are shown in Figure 5. The experimental results show that the more redundant data added the longer the encoding time required, but the encoding time are controlled within 17s.

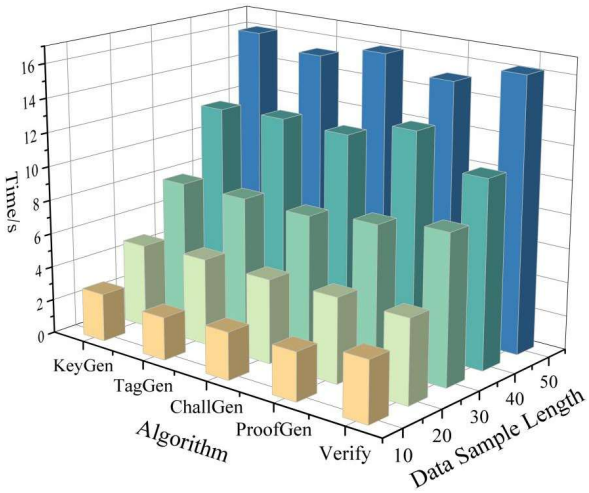

Figure 5: Coding results for data of different lengths

III. B. Validation mechanism optimization effects

In order to prove that the ZSS short signature scheme with the introduction of bilinear pairs and generalized cryptographic hash functions can reduce the total verification time consumption, this subsection compares the total verification time consumption between the original scheme with special hash functions and the ZSS short signature scheme after the introduction of bilinear pairs and generalized cryptographic hash functions.

After experiments, this paper derives the total verification elapsed time of the two schemes after 50 verifications for each of them when different amounts of data are verified at the same time, and the probability of choosing the true seed value for each verification is 50%. The total validation elapsed time for each of the two schemes when 50 pieces of data are validated simultaneously is demonstrated as shown in Fig. 6. After 50 verifications, the total elapsed time of the ZSS short signature scheme with the introduction of bilinear pairs and universal cryptographic

hash function is 4283.94 ms, while the total elapsed time of the original scheme is 6867.36 ms. In comparison, the scheme in this paper reduces the elapsed time by 37.62%.

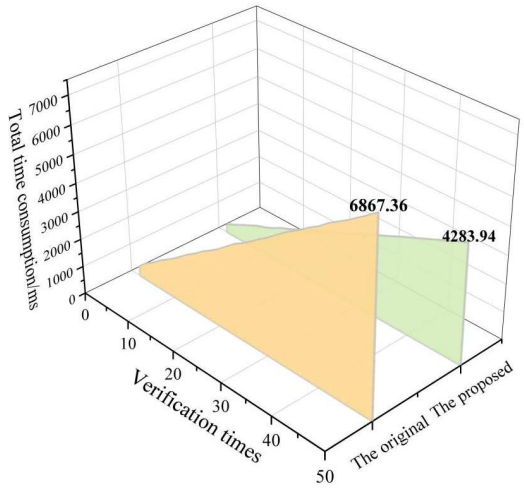

Figure 6: Comparison of total verification time of 50 data simultaneously verified

The total verification elapsed time for each of the two schemes when verifying 500 pieces of data simultaneously is shown in Fig. 7. After 50 verifications, the total time consumed by the ZSS short signature scheme with the introduction of bilinear pairs and generalized cryptographic hash function is 14753.96 ms, while the total time consumed by the original scheme is 29903.05 ms. Compared with the original scheme, the scheme of this paper reduces the time consumed by 50.66%. It can be seen that compared with the original scheme, the scheme in this paper can indeed effectively reduce the total verification elapsed time, and the more the amount of data to be verified at the same time, the larger the proportion of elapsed time reduced.

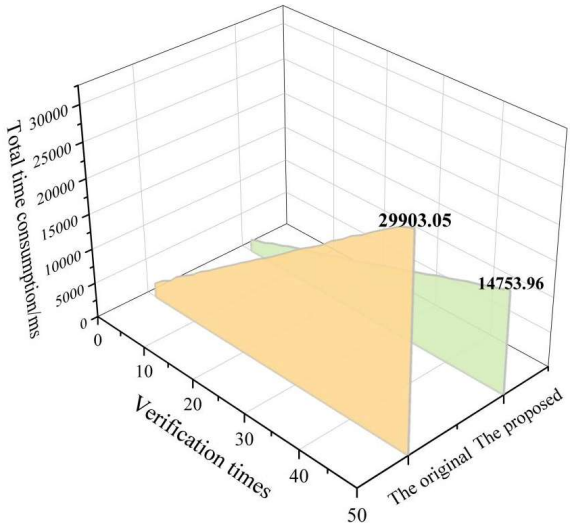

Figure 7: Comparison of total verification time of 500 data simultaneously verified

III. C. Data validation time-consuming results

This section tests the total validation elapsed time from initiating the validation request to getting the validation result under different backup data sizes. A total of four different sizes of data were used for testing, 1MB, 50MB, 500MB, and 1000MB, and the test results are shown in Figure 8. From the test results, it can be seen that when the data size is small, the verification time does not increase significantly. When the data size increases to 500MB or even 1000MB, the verification time starts to increase significantly, and the average verification time for the ten experiments with data sizes of 1MB, 50MB, 500MB, and 1000MB is 601.12s, 644.73s, 987.17s, and 1267.32s, respectively.

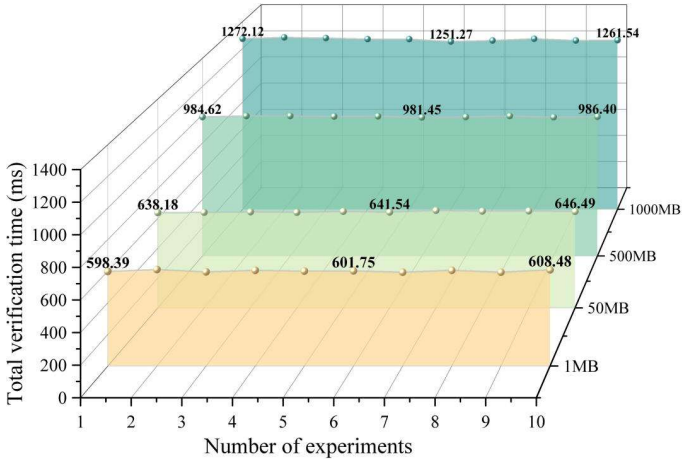

Figure 8: Total time of data verification

The time consumed by TPSP to generate the hash value in the total process of data validation is shown in Fig. 9, and it can be seen that the trend of its time consumption is basically the same as that of Fig. 8. If the time consumed in generating the hash value is compared with the total time consumed in data validation, it can be found that when the data size is small, the smaller the proportion of time consumed in generating the hash value. And as the data size increases, the percentage of time consumed in generating the hash value increases.

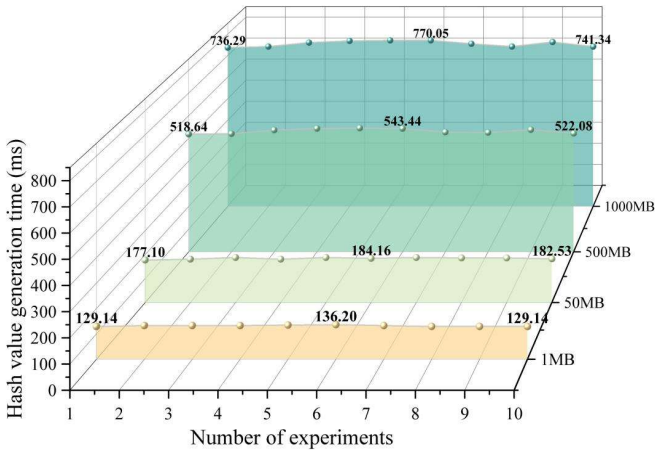

Figure 9: Hash value generation time

IV. Conclusion

In this paper, we design a data integrity verification scheme for accounting cloud platform based on big data analysis, and evaluate its effectiveness through simulation experiments.

All five algorithms can complete the corresponding data processing work in a short time, and the processing time of the algorithms is kept below 30s when the length of the data sample is 50. The more several remaining data are added the longer the coding time required, but the coding time is controlled within 17s.

When verifying 50 pieces of data at the same time, the total elapsed time of the ZSS short signature scheme after the introduction of bilinear pairs and generalized cryptographic hash functions is 4283.94ms, while the total elapsed time of the original scheme is 6867.36ms. In comparison, the scheme in this paper reduces the elapsed time by 37.62%. When verifying 500 pieces of data at the same time, the total time consumed by the ZSS short signature scheme with the introduction of bilinear pairs and generalized cryptographic hash function is 14753.96 ms, while the total time consumed by the original scheme is 29903.05 ms. Compared with the original scheme, the scheme in this paper reduces the time consumed by 50.66%. It can be seen that compared with the original scheme, the scheme in this paper can indeed effectively reduce the total verification elapsed time, and the more the amount of data to be verified at the same time, the larger the proportion of elapsed time reduced.

Tests the total validation elapsed time from initiating a validation request to getting a validation result under different backup data sizes. When the data size increases to 500MB or even 1000MB, the verification time starts to

increase significantly, and the average verification time for ten experiments with data sizes of 1MB, 50MB, 500MB, and 1000MB is 601.12s, 644.73s, 987.17s, and 1267.32s, respectively.The total process of data verification in which the TPSP generates a hash value consumed time trend is basically consistent with the verification time.

Funding

This research is supported by National Social Science Foundation of China (Grant No. 21BGL040).

References

[1] He, Q. (2021, April). Data Mining Analysis Research on Intelligent Application of Cloud Accounting——Taking Cloud Accounting and Financial Sharing Center as an Example. In Journal of Physics: Conference Series (Vol. 1881, No. 4, p. 042061). IOP Publishing.

[2] Wu, H. P., Li, H., & Sun, X. L. (2021). Evolutionary game for enterprise cloud accounting resource sharing behavior based on the cloud sharing platform. IAENG International Journal of Applied Mathematics, 51(1), 1-8.

[3] Vagner, I., Sarakhman, O., & Shurpenkova, R. (2023). Analysis of the development of cloud technologies in accounting. Technology audit and production reserves, 5(4/73), 21-26.

[4] Deng, Y. (2022). Optimising enterprise financial sharing process using cloud computing and big data approaches. International Journal of Grid and Utility Computing, 13(2-3), 272-281.

[5] Tahmid, M. (2023). Accounting in the cloud: a new era of streamlining accounting with cloud technology. Journal of Cloud Computing, 1, 1-14.

[6] Hamundu, F. M., Husin, M. H., Baharudin, A. S., & Khaleel, M. (2020). Intention to adopt cloud accounting: A conceptual model from Indonesian MSMEs perspectives. The Journal of Asian Finance, Economics and Business, 7(12), 749-759.

[7] Ting, W., & Liu, Y. (2020). Design and implementation of intelligent accounting data analysis platform based on industrial cloud computing. EURASIP Journal on Wireless Communications and Networking, 2020(1), 28.

[8] Gupta, D. E. E. P. A. K., & Jain, M. (2017). Impact of cloud accounting on business performance. International Research Journal of Commerce, Arts and Science, 8(12), 321-329.

[9] Ghofirin, M., & Primasari, N. S. (2022). Utilization Of Cloud Accounting During the Covid-19 Pandemic for Owners and Customers in Cooperative Business. IJEMBIS: International Journal of Economics, Management, Business, and Social Science, 2(1), 74-80.

[10] Rawashdeh, A., Rawashdeh, B. S., & Shehadeh, E. (2023). The determinants of cloud computing vision and its impact on cloud accounting adoption in SMBs. Human Behavior and Emerging Technologies, 2023(1), 8571227.

[11] Wang, T., & Liu, Y. (2020). Design and implementation of intelligent accounting data analysis platform based on industrial cloud computing. EURASIP Journal on Wireless Communications and Networking, 2020(1).

[12] Duan, Y. (2024). Design of Accounting Information Big Data Analysis Platform Based on Cloud Computing. Procedia Computer Science, 247, 1128-1136.

[13] Dai, Q. (2022). Designing an accounting information management system using big data and cloud technology. Scientific Programming, 2022(1), 7931328.

[14] Chen, Y. (2021). Framework of the smart finance and accounting management model under the artificial intelligence perspective. Mobile Information Systems, 2021(1), 4295191.

[15] Achar, S. (2018). Security of accounting data in cloud computing: a conceptual review. Asian Accounting and Auditing Advancement, 9(1), 60-72.

[16] Atadoga, A., Umoga, U. J., Lottu, O. A., & Sodiya, E. O. (2024). Evaluating the impact of cloud computing on accounting firms: A review of efficiency, scalability, and data security. Global Journal of Engineering and Technology Advances, 18(2), 065-074.

[17] Wan, J. (2023, July). Cloud Data Integrity Verification Algorithm for Accounting Informatization Under Sharing Mode. In International Conference on Frontier Computing (pp. 317-322). Singapore: Springer Nature Singapore.